robotron

Programmtechnische
Beschreibung

REDAPROG

H 0213-2223-1 M 0030

Büro- und Personalcomputer Betriebssystem SCP

Programmtechnische
Beschreibung

POS

R E D A P R O G

Dienstprogramm fuer die
Behandlung von
REDABAS-Programmen

SCP

VEB Robotron-Projekt Dresden

H 0213-2223-1 M0030

Die vorliegende Systemunterlagen-Dokumentation REDAPROG entspricht dem Stand von 10/86

Nachdruck, jegliche Vervielfaeltigung oder Auszuege daraus sind unzulaessig.

Die Ausarbeitung erfolgt durch ein Kollektiv des
VEB Robotron-Projekt Dresden

Herausgeber:
VEB Robotron-Projekt Dresden
Leningrader Str. 9, Dresden, 8010

Das Programmpaket REDAPROG und diese Dokumentation werden Vertrieben durch

- VEB Robotron-Projekt Dresden
  Bereich K

  Leningrader Str. 9
  Dresden
  8010

Ag 706/298/87 III/21/19

Kurzreferat

Zur Anwenderdokumentation von REDAMASK gehoeren die Schriften

- Anwendungsbeschreibung
- Programmtechnische Beschreibung

Die Anwendungsbeschreibung enthaelt

- eine Kurzcharakteristik von REDAMASK bezueglich Leistungsfaehigkeit und Handbuch
- Aussagen ueber Anwendungsbereich und Voraussetzungen fuer die Nutzung.

Die Programmtechnische_Beschreibung enthaelt detaillierte Angaben ueber die Arbeitsweise, die Funktionen, sowie eine Zusammenstellung aller REDAPROG-Kommandos mit ausfuehrlicher Beschreibung.

Inhaltsverzeichnis Seite

Einfuehrung 5

1. Lieferumfang 5

2. Voraussetzungen fuer die Anwendung 5

3. Handhabung von REDAPROG 6
3.1. Das Anfertigen einer Sicherheitskopie 6
3.2. Uebernahme des Betriebssystems 6
3.3. Installieren 6
Durchfuehrung der Installierung 7
Testen der Steuercodes 7
Eigeninstallation 8
Fehlernachrichten 9
3.4. Starten und Beenden von REDAPROG 10
3.5. Arbeiten mit REDAPROG 11
3.6. Das REDAPROG - Menue 11
3.6.1. Direktkommandos 12
3.6.2. Spezifizieren von Dateinamen 12
3.6.3. Festlegen der Umgebung 15
3.6.4. Gestaltung der Ausgabedateien 16
3.6.5. Funktionen 17

4. REDAPROG - Dienste 20
4.1. Laufzeitverkuerzung von REDAPROG -Programmen 20
4.2. Verringerung des Programmieraufwandes 23
4.3. Vereinfachung der Fehlersuche und Wartung 24
4.3.1. Die verbesserte Gestaltung der Programmdokumentation 24
4.3.2. Das automatische Erstellen der Programmdokumentation 27
4.3.3. Das automatische Erstellen von Sicherheitskopien 29

5. Leistungsparameter 30

6. Fehlernachrichten 30

7. REDAPROG - Kommandos 26

Sachwortverzeichnis 44

Einfuehrung

Die Leistungsfaehigkeit, Flexibilitaet und relativ leichte Handhabbarkeit des Relationalen Datenbankbetriebssystems REDABAS fuehrte zu einer Viel-zahl und Vielfalt von Einsatzfaellen.

Um die Vorteile, die der Einsatz von REDABAS bietet, optimal nutzen zu koennen und darueberhinaus zu verbessern, wurde REDAPROG als Zusatzkom ponente der 8-Bit-Version von REDABAS entwickelt.

REDAPROG unterstuetzt den REDABAS-Programmierer durch folgende Dienste:

- Erhoehung der Verarbeitungsgeschwindigkeit von REDABAS-Anwenderprogrammen
- Verringerung des Programmieraufwandes
- Vereinfachung der Fehlersuche und Wartung

REDAPROG ist leicht zu handhaben, seine Anwendung setzt die Kenntnisse von REDABAS voraus.

1. Lieferumfang

Der Lieferumfang von REDAPROG umfasst folgende Bestandteile:

- Programmkomponenten auf Diskette
  REDAPROG.COM
  RPROGKOM.DAT
  RPROGKEY.TXT
  Beispieldateien
- Anwenderbeschreibung
- Programmtechnische Besschreibung
- Anwendungsbeschreibung

2. Voraussetzung fuer die Anwendung

Hardware

Mindestkonfiguration
- Personalcomputer ab 64 Kbytes RAM
  (auf Basis Mikroprozessor U880)
- 1 Diskettenlaufwerk fuer 8"-Standarddisketten oder
  5,25"-Minidisketten
- 1 Bildschirm (mindestens 16 Zeilen x 64 Zeichen)
- 1 Drucker (wahlweise)

Das Kombinat Robotron bietet fuer die Arbeit mit REDAPROG die Buerocomputer der Erzeugnisreihen A5110, A5120 und A5130 sowie den Personalcomputer robotron 1715 und deren variable

Peripherie an.

Software

- Betriebssystem SCP oder kompatible Betriebssysteme

- Relationales Datenbankenbetriebssystem REDABAS

Vom REDAPROG-Nutzer werden keine speziellen EDV-Kenntnisse erwartet. Voraussetzung ist jedoch die Kenntnis von REDABAS und SCP

3. Handhabung von REDAPROG

3.1. Das Anfertigen einer Sicherheitskopie

Bevor Sie erstmalig mit REDAPROG arbeiten, fertigen Sie sich unbedingt ein Dublikat der REDAPROG-Diskette an. Damit wird verhindert, dass bei einer unbeabsichtigten Zerstoerung des Disketteninhaltes (z. B. Geraete oder Bedienfehler) die Originalprogrammkomponennten verlorengehen

3.2. Uebernahme des Betriebssystems

Bitte beachten Sie, dass die REDAMASK-Diskette im Auslieferungszustand nicht das Betriebssystem enthaelt. Um REDAMASK zu nutzen, muesen Sie entweder als erstes eine Diskette, die das Betriebssystem enthaelt, in ein Laufwerk des Computers einlegen, oder Sie kopieren das Betriebssystem auf Ihre REDAMASK-Diskette. Letzteres wird empfohlen, da die erforderliche Anzahl an Spuren auf jeder Diskette fuer das Betriebssystem freigehalten wird, also kein zusaetzlicher Platz benoetigt wird. Es vereinfacht die Handhabung, wenn sich Beriebssystem und REDAMASK auf einer Diskette befinden.

3.3. Installieren

Sowohl Rechner als auch Bildschirm nutzen fuer die Bildschirmanzeige unterschiedliche Steuercodes. Deshalb ist es notwendig, REDAPROG mitzuteilen, welche Codes Ihr spezielles System vewenden soll Fuer diese sogenannte Installierungsprozedur steht die Programmkomponennte RPROGINS.COM zur Verfuegung (nach erfolgreicher Installierung koennen Sie RPROGINS.COM von der Kopie Ihrer REDAPROG-Diskette entfernen).

Standardmaessig ist REDAPROG auf den Bildschirm BAB2 mit 1920 Zeichen (24 Zeilen x 80 Spalten) angepasst

Durchführung der Installation

Wenn Sie sich in der SCP-Kommandoebene befinden, tippen Sie

RPROGINS

Ein. Druecken Sie anschliessend die <ET>-Taste. (Beachten Sie bitte, dass sich die Datei RPROGKON.DAT auf der Diskette im aktuellen Laufwerk befinden muss, sonst erscheint bei einer speater gewuenschten permanenten Installierung eine Fehlernachricht.)

Auf dem Bild erscheint das Installierungsmenue. Sie waehlen Ihren Bildschirm durch Eingabe des entsprechenden Buchstaben aus (Sie koennen auch eine sogenannte Eigeninstallation vornehmen. Dieser Vorgang wird spaeter beschrieben.)

Nachdem Sie sich von der Richtigkeit Ihrer Wahl ueberzeugt haben werden Sie aufgefordert:

WAEHLEN SIE DIE NAECHSTE FUNKTION!

Sie haben 5 Auswahlmoeglichkeiten (A...E) mit folgenden Funktionen:

A Wahlen Sie A, so bedeutet das, dass Sie den von Ihnen im Installierungsmenue gewaehlten Bildschirm permanent installieren wollen. Es empfiehlt sich, bevor Sie A und anschliessend E (Verlassen von RPROGINS) druecken, die Nachfolgend beschrieben Testprozedur (Druecken von B) durchlaufen zu lassen.

B ermoeglicht das testen der Steuercodes des Bildschirms, den Sie im Installierungsmenue ausgewaehlt haben, bevor Sie sich durch Druecken von A fuer eine permanente Installierung entscheiden. Im naechsten Abschnitt wird Dieser Test ausfuehrlicher beschrieben

C gestattet es, die Installierungsprozedur zu wiederholen.

D Wenn Sie D waehlen, erhalten Sie eine Uebersicht der Steuercodes des von Ihnen gewaehlten Systems.

E Mit der Wahl von E beenden Sie die Installierung.

Der Normalfall einer Installierung wird sein, dass Sie nach erfolgreich ausgefuehrtem Test der Steuercodes (B) A druecken und durch Druecken von E die Installierung beenden

Tasten der Steurcodes

Haben Sie sich für den Test der Steuercodes entschieden, werden Sie aufgefordert:

WAEHLEN SIE EINE TESTMOEGLICHKEIT

Sie haben 3 Auswahlmoeglichkeiten (A, B, 2).

A Bildschirm loeschen: Druecken Sie (nachdem Sie A gedrueckt haben) irgendeine Taste, muss der Bildschirm voellig leer sein. Wenn dies nicht so ist, sind Ihre Steuercodes fuer das Loeschen des Bildschirms falsch. Das nochmalige Druecken einer Taste beendet den Test.

B Cursordarstellung: Der Test der Cursordarstellung setzt einen erfolgreichen Test des Bildschirmloeschen voraus. Nach der Anzeige einer kurzen Beschreibung des Tests druecken Sie wieder eine beliebige Taste. Auf dem Bildschirm erscheint die folgende Uebersicht der Einteilung von Zeilen und Spalten

```
0.........1.........2.........3.........4.........5
 1
 2   --- DAS IST ZEI. 2 ---
 3
 4
 5
 6
 7                                                  S
 8                                                  P
 9  * (ZEI. 9, SP. 4)                               .
10
11                                                  5
12                                      * (12,39)   0
13
14
15
```

Erscheint das Bild nicht in der dargestellten Form, sind die Cursoragressierungscodes falsch. Es gilt auch hier, bei nochmaligem Druecken einer taste endet dieser Test.

Z Testende:
Durch Druecken von Z wird das Testen der Steuercodes beendet

Eigenistallation

Ist Ihr Bildschirm nicht auf den Installierungmenue aufgelistet oder funktionieren Ihre Steuercodes nich wie gewuenscht, haben Sie die Moeglichkeit, eine individuelle Installierung vorzunehmen. (Naehere Informationen ueber die Steuercodes erhalten Sie aus dem Handbuch Ihres Bildschirms.)

Ausgangspunkt einer Eigeninstallation ist das Installierungsmenue. Sie waehlen den Buchstaben Z.

Nach einer Starmeldung werden Sie nach dem sogenannten Einleitungszeichen gefragt

Das Einleitungszeichen ist ein spezielles Zeichen, mit dem die Steuerfolgen beginnen, die an den Bildschirm gesendet werden.

Nicht jeder Bildschirm verwendet ein solches Einleitungszeichen. Druecken Sie nur <ET>, bringen Sie zum Ausdruck, dass Ihr Bildschirm kein Einleitungszeichen nutzt

Sie beginnen mit der Definition der Steuercodes fuer das Bildschirmloeschen

Haben Sie sich eingangs generell fuer ein Einleitungszeichen entschieden, werden Sie zunaechst wieder gefragt, ob Sie ein Einleitungszeichen speziell fuer das Bildschirmloeschen brauchen. Anschließend definieren Sie die Zeichenfolge (max. 2 Zeichen) fuer das loeschen des Bildschirms.

Danach erfolgt die Definition der Cursoradressierung. Haben Sie eingangs die Frage nach dem Einleitungszeichen positiv beantwortet, werden Sie jetzt gefragt, ob die Cursoradressierung ein Einleitungszeichen verlangt.

Anschliessend (oder wenn kein Einleitungszeichen gewaehlt wurde, sofort) muessen Sie das Cursoradressierungszeichen und die Anordnung von Zeilen- und Spaltenposition eingeben. Soll die Zeile vor der Spalte gesendet werden, waehlen Sie Y; im umgekehrten Fall X.

Danach erfolgt die Definition der Konstanten, die den Zeilen- und Spaltennummern bei der Cursorpositionierung hinzugefuegt werden soll. Gibt es eine Konstante, geben Sie deren Dezimalwert ein. Gibt es keine, druecken Sie <ET>.

Nach dieser Eingabe werden Sie augefordert:

WAEHLEN SIE DIE NAECHSTE FUNKTION!

Es empfiehlt sich nun das Testen der von Ihnen eingegebenen Steuercodes durch Druecken von B, wie es im vorigen Abschnitt beschrieben wurde

Sind die Tasts erfolgreich verlaufen, installieren Sie durch Druecken von A Ihre Version permanent. Gibt es Unstimmigkeiten oder Fehlerausschriften, lesen Sie im folgenden Abschnitt nach und korrigieren Sie entsprechend!

Fehlernachrichten

Fehlernachricht:	Beschreibung:
SYSTEM IST NICHT VERFÜGBAR!.. NEUE EINGABE:	Das von Ihnen ausgewaehlte system ist im Installie-Rungsprogramm noch nicht Vorhanden. Entweder Sie waehlen ein andres System oder Sie fuehren eine Eigeninstallation durch!
FALSCHE EINGABE!.. NEUE EINGABE:	Sie haben ein ungültiges Zeichen eingegeben (Gueltig sind A, B oder Z)

RPROKON.DAT NICHT VORHANDEN!	Ihre Konfigurationsdatei (RPROKON.DAT) befindet sich entweder nicht auf der Diskette im aktuellen Laufwerk oder ist zerstoert worden Die von Ihnen gewuenschte permanente Installierung kann nicht ausgefuehrt werden. Kopieren Sie die Datei erneut von einer Sicherheitsdiskette!
SCHREIBFEHLER RPROKON.DAT!	Entweder ist Ihre Diskette voll oder Ihre Konfigurationsdatei ist nicht in Ordnung Ueberpruefen Sie!

3.4. Starten und Beenden von REDAPROG

Unter Beachtung der in den vorangegangenen Abschnitten gegebenen Hinweise koennen Sie REDAPROG starten. Fuehren Sie bitte folgende Handlung aus:

- Schalten Sie den Rechner ein und legen Sie die REDAPROG-Diskette ins Laufwerk A.
- Das Betriebssystem meldet seine Dialogbereitschaft durch die Anzeige von "A>" auf dem Bildschirm. War SCP bereits vorher aktiv, koennen Sie die REDAPROG-Diskette auch in ein beliebiges anderes Laufwerk einlegen. Dieses Laufwerk muss vorher als aktuelles Laufwerk zugewiesen sein. Starten Sie REDAPROG nicht vom aktuellen Laufwerk, kommt es zum Programmabbruch.
- Sie tippen REDAPROG ein und druecken die <ET>-Taste (REDAPROG wird in den Arbeitsspeicher geladen).
- Nach erfolgreichem Start meldet sich REDAPROG mit der Ausschrift

  ```
  ROBOTRON-PROJEKT DRESDEN

  **************
  *  REDAPROG  *
  **************

  Version 1.0 - 01.10.86
  .
  .
  .
  BITTE FORTSETZEN MIT  <ET>!
  ```

- Nach dem Druecken der <ET>-Taste wird das REDAPROG-Menue angezeigt,Ihre eigentliche Arbeit mit REDAPROG beginnt.
- Das Beenden von REDAPROG geschieht durch Eingabe der Zahl

99 und anschliessendes druecken von <ET>.
REDAPROG verabschiedet sich mit der Ausschrift:

ENDE DER AUSFUEHRUNG

3.5. Arbeiten mit REDAPROG

REDAPROG-Funktionen werden durch Kommandos realisiert. Generell gibt es zwei Moeglichkeiten, diese Kommandos abzurufen.

Eine Moeglichkeit besteht darin, REDAPROG-Kommandos direkt in die REDABAS-Befehlsdatei (Dateityp .PRG) einzufuegen. Eine dem Kommando vorangestellte Zeichenfolge * % kennzeichnet die REDAPROG-Kommandos fuer REDABAS als Kommentare. Diese Kommandos werden bei der Befehlsabarbeitung durch REDABAS ignoriert und bei einem REDAPROG-Lauf zur weiteren Verarbeitung annalysiert.

Die zweite Moeglichkeit, REDAPROG-Funktionen zu aktivieren, besteht darin, sie ueber das REDAPROG-Menue abzurufen. Dieses Menue erscheint nach dem REDAPROG-Startbild, es hat das im Abschnitt 3.6. gezeigte Aussehen.

Mit den Angaben im REDAPROG-Menue veranlassen Sie eine Vor-einstellung der Funktionen, mit denen sie die meiste Zeit arbeiten moechten. Diese voreingestellten Werte koennen jedoch durch Einfuegen von REDAPROG-Kommandos in eine REDABAS-Befehlsdatei veraendert werden. Das bedeutet, REDAPROG-Kommandos haben gegenueber den im Menue gewaehlten Funktionen Vorrang.

3.6. Das REDAPROG-Menue

```
                              REDAPROG  M E N U E
*============================================================*
*          EING.DATEIEN                      AUSG.DATEIEN     *
*-------------------------          ------------------------- *
*(1)  QUELLE-MENUE2.PRG              (4) ZIEL   -MENUE2Z.AUS   *
*(2)  WOERTER-RPROGKEY.TXT           (5) STRUK  -STRUK.STR     *
*(3)  GRUPPE-.DAT                    (6) SUBMIT -LISTE.SUB     *
*============================================================*
*(10) 1.LAUFWERK-A:                 (14) QUELL-/ZIELTYP-PRG,AUS*
*(11) 2.LAUFWERK-B:                 (15) AUSGABEEINHEIT-LST:   *
*============================================================*
*(12) BEFEHLSTABULATOR-4            (16) MAX.DO-STUFE      -7  *
*(13) STRUKTURTAB.    -5            (17) MAX.INCLUDE-STUFE -7  *
*============================================================*
*(20) include  (21) skip   (22) conv    (23) stru  (24) code *
*(25) display  (26) do     (27) process (28) page  (29) sub  *
*(30) NOgrup   (31) return (32) comment (33) use             *
*============================================================*
 (0) AUSFUEHREN (34) SICHERN (99) ENDE
* * * * * BITTE AUSGEWAEHLTE NR. EINGEBEN :
```

Das REDAPROG-Menue ist funktionell in 5 Abschnitte unterteilt, die nach folgend beschrieben werden. Sie koennen jede der im REDAPROG-Menue angefuehrten Moeglichkeiten oder Funktionen durch Eingabeder entsprechenden Nummer ansprechen.

Waehrend Sie in den oberen drei Abschnitten Voreinstellwerte belassen oder veraendern koennen, werden Ihnen im vierten Abschitt Funktionen angeboten, die Sie in der angezeigten Form uebernehmen oderausser Kraft setzen koennen. Lediglich die im letzten Abschnitt angebotenen "Direktkommandos" aktivieren Sie direkt durch Eingabeder Zahlen.

Waehlen Sie eine nicht im Menue angefuehrten Zahl, erscheint dieAusschrift:

*** FALSCHE AUSWAHL ***

3.6.1. Direktkommandos

In der letzten Zeile des REDAPROG-Menues sind die drei REDAPROG-Direktkommandos angefuehrt. Sie entscheiden durch die Auswahl der entsprechenden Zahl, ob Sie

- die Verarbeitung entsprechend der von Ihnen gewaehlten Moeglichkeiten und Funktionen (Zahl (1) bis (33)) ausfuehren moechten (0)
- die von Ihnen gewaehlten Moeglichkeiten und Funktionen fuer kuenftige Verarbeitung sichern wollen (34)
- REDAPROG beenden und zum Betriebssystem zurueckkehren moechten (99)

Die Eingabe der von Ihnen gewuenschten Zahl wird mit dem Druecken der <ET>-Taste abgeschlossen.

3.6.2. Spezifizieren von Dateinamen

Der obere Abschnitt des REDAPROG-Menues beinhaltet die Namen der mit REDAPROG zu bearbeitenden bzw. von REDAPROG erstellten Dateien.

Nach Eingabe einer der Ziffern 1 bis 6 in der Dialogzeile koennen Sie Dateinamen neu vergeben bzw. aendern. Ein Dateiname darf max. 8 Zeichen lang sein und muss mit einem Buchstaben beginnen.

Haben Sie eine Ziffer gewaehlt, fordert Sie REDAPROG mit der Ausschrift:

NEUEN NAMEN EINGEBEN:

entsprechend auf.

Geben Sie hier nicht den Dateityp an. Wenn Sie den Dateityp von (1)QUELLE oder (4) ZIEL gegenueber den voreingestellten Werten (.PRG bzw. .AUS) veraendern wollen, geschieht das bei (14) QUELL-/ZIELTYP. Die anderen Dateitypen -(2).TXT, (3).DAT, (5).STR, (6).SUB koennen nichtveraendert werden.

Eingabedateien
---

(1) QUELLE — Es ist der Name der REDABAS-Befehlsdatei anzugeben, die mit REDAPROG bearbeitet werden soll.Auch wenn diese Befehlsdatei weitere Programmaufrufe enthaelt, ist nur der Name der rufenden Datei zu spezifiziren.
Voreinstellung fuer diese Datei ist die REDABAS/Beispieldatei MENUE2.PRG. (vergl. REDABAS-Programmtechnische Beschreibung Bild 62).

2) WOERTER — Hier kann man den Namen einer Textdatei angeben,die all die Woerter enthalten,die mit der (22)conv Funktion in Grossbuchstaben umzuwandeln sind.
Die in der spezifizierten Datei enthaltenen zw.aufzunehmenden Schluesselwoerter muessen alphabetisch sortiert und klein geschrieben sein.
Es sind max.150 Schluesselwoerter moeglich.
Voreinstellung fuer die WOERTER-Datei ist PROGKEY.TXT, die die REDABAS-Befehlssteuerwoerter enthalten. Mit Hilfe von RPROGKEY.TXT kann die Lesbarkeit von REDABAS-Programmen verbessert werden (vg). Abschnitt 4.3.1.), wenn die Funktion (22)conv aktiviert wurde.
Diese Datei wird von REDPROG nicht veraendert,sie kann mit einem Textverarbeitungsprogramm, REDABAS ,oder dem Editor von SCP gewartet werden.
Die Datei wird auf einem der unter (10) Und (11) genannten Laufwerke erwartet.

(3) GRUPPE — Es kann der Name einer Datei angegeben werden, die eine Anzahl von Dateinamen (ohne Angabe des Dateityps) enthaelt, die mit der spaeter beschriebenen (30) group-Funktion behandelt werden.
Fuer die Erstellung dieser Datei ist der Nutzer selbst verantwortlich. Sie wird nicht von REDAPROG gepflegt und kann mit einem Textverarbeitungssystem, REDABAS oder dem Editor von SCP gewartet werden.
Der Dateityp ist mit .DAT definiert.

(4) ZIEL

Es wird der Name einer Datei angegeben, die nach entsprechender Bearbeitung durch REDAPROG erstellt werden soll und eine ausfuehrbare REDABAS-Befehlsdatei darstellt.
Voreinstellung fuer diese Datei ist MENUE2Z.AUS, eine Textdatei, die lediglich die Arbeit von REDAPROG veranschaulichen soll. In spaeter folgenden Demonstrationsbeispielen sehen Sie, wie durch die REDAPROG Behandlung aus der Eingabedatei MENUE2.PRG die Ausgabedatei MENUE2Z.AUS wird.

(5) STRUK

Es wird der Name einer Datei angegeben, die von REDAPROG nur dann erstellt wird, wenn die (23)stru- und(26)do-Funktion aktiviert werden, und die Funktion (30) auf NOgrup gesetzt ist.
Die Datei enthaelt ein Diagramm der Baumstruktur der REDABAS-Befehlsdateien, um darzustellen, wo welche Unterprogramme in einem Hauptprogramm aufgerufen werden.
(Eine Spezifizierung der (5)STRUK-Datei setzt die Spezifizierung einer (4)ZIEL-Datei voraus.)
Voreinstellung fuer die STRUK-Datei ist STRUK.STR.

(6) SUBMIT

Hier kann der Name einer Datei angegeben werden, die von REDAPROG erstellt wird, wenn die (29)sub-Funktion aktiviert wurde. (Voraussetzung fuer diese Funktion ist ausserdem ie ktivierung von (23)stru und (26)do.)
Die SUBMIT-Datei kann mit dem SCP-Kommando SUBM bearbeitet werden.
Sie enthaelt SCP-Kommandos, die Ihre Programme in derselben Anordnung, wie sie in der (5)STRUK-Datei angeordnet sind, als Druckliste oder Diskettendatei ausgegeben. Sie koennen mit der REDAPROG-Funktion (29)sub somit eine komplette Programmuebersicht auf einer Druckliste erzeugen.
(Eine Spezifizierung der (6)SUBMIT-Datei setzt die Spezifizierung der (4)ZIEL-Datei und (5)STRUK-Datei voraus.)
Voreinstellung fuer die SUBMIT-Datei ist LISTE.SUB.

3.6.3. Festlegen der Umgebung

Im zweiten Abschnitt des RDAPROG-Menues bestimmen Sie Ihre Arbeitsumgebung
Wollen Sie von den Voreingestellwerten abweichen, geben Sie in der Dialogzeile eine der Ziffern 10, 11, 14 oder 15 ein.
REDAPROG fordert Sie auf:

BITTE EINEN NAMEN EINGEBEN:

Tippen Sie Ihren gewünschten Wert ein und schliesen Sie die Eingabe mit <ET> ab!

(10) 1. Laufwerk — Hier wird die Bezeichnung des Laufwerkes angegeben, auf dem REDAPROG die Ausgabedatei anlegen soll. Der entsprechende Platz muss auf der dort befindlichen Diskette vorhanden sein, sonst kommt es zum Programmabbruch.
Ausserdem erwartet REDAPROG auf diesem oder dem unter (11)2.LAUFWERK spezifizierten Laufwerk die Eingabedateien, einschliesslich der unter (2) genannten Woerter-Datei.
Voreinstellung ist A:.

(11) 2. Laufwerk — Sie haben die Moeglichkeit, fuer die Eingabedateien alternativ ein zweites Laufwerk zu spezifiziern. Wird die Eingabedatei nicht auf dem bei (10) 1. LAUFWERK spezifizierten Laufwerk gefunden durchsucht REDAPROG automatisch die auf dem 2. Laufwerk befindliche Diskette.
Voreinstellung ist B:.

(14) QUELL-/ZIELTYP — Hier kann - durch ein Komma getrennt - der Dateityp fuer folgende Ein-und Ausgabedateien spezifiziert werden:

Moegliche Eintragungen vor dem Komma:
1) der Dateityp der bei (1)QUELLE spezifizierten Datei
2) der Dateityp der Eingabedatei, deren Name in der bei (3)GRUPPE spezifizierten Datei stehen (nicht der Dateityp der GRUPPE-Datei selbst!)

Moegliche Eintragungen nach dem Komma:
1) der Dateityp der bei (4)ZIEL spezifizierten Datei
2) der Dateityp der Ausgabedateien, deren Name in der bei (3) GRUPPE spezifizierten Datei stehen und die mit der group-Funktion bearbeitet wurden.

Wird die group-Funktion verwendet, muss die Angabe vor dem Komma und die Angabe nach dem Komma unterschiedlich sein. Generell besteht der Dateityp aus 3 Buchstaben.
Voreinstellung ist PRG.AUS.

(15) AUSGABEEINHEIT Das ist der logische SCP-Geraetename, den Sie fuer die Ausgabedatei waehlen, die bei der Abarbeitung der bei (6)SUBMIT definierten Datei entsteht.
Die Voreinstellung ist .LST:
Sie koennen aber auch einen anderen logischen Geraetenamen, den SCP zulaesst z.B. PRN - angeben.
Wollen Sie Sicherheitskopien Ihrer Dateien erzeugen, (vgl. Abschnitt 4.3.3.), koennen Sie hier das Laufwerk spezifizieren, auf dem sich die Diskette befindet,die die Sicherheitskopie aufnehmen soll.

3.6.4. Gestaltung der Ausgabedateien

Im dritten Menueabschnitt werden Zahlenwerte definiert, die ueber Inhalt und Form der mit REDAPROG erstellten Ausgabedateien bestimmen.

(12) BEFEHLSTABULATOR Der hier angegebene "Tabulierungsfaktor" bezeichnet die Anzahl an Spalten in den REDABAS-Befehlsdateien, um die die Befehle eingerueckt werden, die durch IF/ENDIF-, DO/ENDDO und CASE/ENDCASE-Befehle eingeschlossen sind.
Die Voreinstellung ist 4, d. h. es werden Tabulatorabstaende von jeweils 4 Zeichen gesetzt.

(13) STRUKTURTAB. Hir wird die Spaltenanzahl angegeben, die als "Tabulierungsfaktor" fuer das Einruecken der in der Strukturdatei enthaltenen Befehlsdateinamen gewaehlt wird. Die Voreinstellung ist 5.

(16) MAX. DO-STUFE Es kann die Anzahl von Verschachtelungsebenen in der Struktur einer Befehlsdatei angegeben werden, die REDAPROG verarbeiten soll.
Geben Sie beispielsweise 4 an, werden die Dateien ab der 5. Ebene nicht aufgeloest.
Jede mit "DO <befehlsdatei>" aufgerufene Befehlsdatei stellt eine Verschachtellung dar.
REDAPROG gestattet maximal 8 Verschachtelungsebenen.

Voreinstellung ist 7.

(17) MAX. INCLUDE-STUFE — Es kann die Anzahl der Verschachtelungsebenen in einer Serie von INCLUDE-Befehlen in Befehlsdateien angegeben werden. [entspricht prinzipiell (16)] REDAPROG erlaubt auch hier maximal 8 Verschachtellungsebenen. Voreinstellung ist 7.

3.6.5. Funktionen

Im vierten Abschnitt des Menues werden Ihnen alle REDAPROG-Funktionen angeboten. Die hier getroffene Auswahl bestimmt bei der darauffolgenden Bearbeitung einer Befehlsdatei durch REDAPROG die auszufuehrenden Leistungen.

Es obliegt Ihrer Entscheidung, welche von diesen Funktionen Sie auf ihre REDABAS-Befehlsdatei anwenden wollen. Entweder Sie uebernehmen die Funktion in der angezeigten Form oder Sie waehlen das Gegenteil, indem Sie eine der Zahlen 20 bis 33 eintippen.

(Z. B. ist im Menue (33) use die Voreinstellung. Entscheiden Sie Sich fuer NOuse, tippen Sie die Zahl 33 ein. Mit dem nochmaligen Eintippen von 33 koennen Sie den Ausgangszustand wiederherstellen.)

Bitte beachten Sie folgendes :
Sie koennen diese Funktion durch das Hauptmenue aktivieren und/oder durch einfuegen spezieller Kommentare in Ihre REDABAS-Befehlsdatei (vgl.Kapittel 6) aufrufen.

Haben Sie die Funktionen (23) stru, (24) code und (29) sub im Menue auf NOstru, Nocode oder NOsub gesetzt, duerfen Sie sie nicht in Ihre Befehlsdatei aktivieren! Es erfolgt keine Ausgabe, die Ausfuehrung von REDAPROG wird abgebrochen. Es erscheint die Fehlermeldung "ENDE DER AUSFUEHRUNG".

Nachfolgend erhalten Sie eine Kurzbeschreibung der Funktionen. Eine detallierte Erklaerung erfolgt bei der Beschreibung der REDAPROG-Kommandos im Kapittel 7.

(20) include — Diese Funktion ermoeglicht das Einfuegen von Programmen oder Programmteilen in REDABAS-Befehlsdateien. Die Programme muessen als Textdateien vorliegen und sollten den Dateityp .INC haben.

(21) skip — skip ermoeglicht das einfuegen von Leerzeilen in Befehlsfolgen der Zieldatei.

(22) conv — Mit conv koennen Sie in Kleinbuchstaben vorliegende Befehls-Steuerwoerter Ihrer REDABAS-Programmdatei oder beliebige andere Begriffe in Grossbuchstaben um-

wandeln. Die umzuwandelnden Woerter sind in der unter (2) WOERTER spezifizierten Datei zu speichern.

(23) stru — Wird diese Funktion gewaehlt, erstellt REDAPROG eine Ausgabedatei (vgl. (5)STRUK), in der die Baumstruktur Ihrer REDABAS-Befehlsdateien aufgelistet ist. Voraussetzung fuer diese Funktion ist, dass die do-funktion (26) gewaehlt wurde und keine Dateigruppe verarbeitet werden soll (d. h. es existiert keine Datei (3) GRUPPE und es ist (30) NOgroup gesetzt).

(24) code — Mit code werden die Ausgabe-Befehlsdateien erstellt.
Ist beispielsweise die (26)do-Funktion aktiviert, wird eine erweiterte Befehlsdatei erstellt, d. h. anstelle der Unterprogrammaufrufe wird der gesamte Befehlsumfang der Unterprogramme eingefuegt.
Ist die (30)group-Funktion aktiviert, entsteht eine Gruppe von Befehlsdateien.
Wurde NOcode gewaehlt, werden diese Ausgabedateien durch REDAPROG nicht erstellt.

(25) display — display ermoeglicht die Anzeige der bearbeiteten Programmzeilen und der REDAPROG - Nachrichten, wenn (0) AUSFUEHREN aktiviert wurde

(26) do — Ist die do-Funktion aktiviert, werden die in einer REDABAS-Befehlsdatei vorhandenen Unterprogrammaufrufe (DO <befehlsdatei>) durch den gesamten Befehlsumfang der Unterprogramme ersetzt.
(26)do und (24) code bedingen einander ,d.h. wenn do aktiviert wird, muss auch code aktiviert sein.

(27) process — Diese Funktion veranlasst die Abarbeitung der REDAPROG-Kommandos in den REDABAS-Befehlsdateien.

(28) page — Mit page haben Sie die Moeglichkeit, Seitenvorschub zu veranlassen.

(29) sub — Diese Funktion erzeugt eine Ausgabedatei fuer die Nutzung des SCP-Kommandos SUBM. Diese Ausgabedatei muss bei (6) SUBMIT und (15) AUSGABEEINHEIT definiert sein. Das Aktivieren von sub setzt die Einstellungen (23) stru, (24) code, (26)do und (30)NOgroup voraus.

(30) Nogroup

Im Menuebild ist diese Funktion bewusst durch das Praefix NO ausgeschaltet worden, da dies der Normalfall einer REDAPROG-Anwendung sein wird.

(30) group

Wird davon abweichend group gewaehlt, so bewirkt dies, dass die in einer Gruppe zusammengefassten Befehlsdateien mit REDAPROG behandelt und einzeln ausgegeben werden. Anstelle einer Ausgabedatei entstehen mehrere separate Ausgabedateien.

Als Eingabedatei fungiert die Gruppe der Befehlsdateien ,deren Name (ohne Angabe des Dateityps!) in der unter (3)GRUPPE spezifizierten Datei aufgelistst sind. Fuer den Aufbau und die Wartung dieser Datei ist der Nutzer selbst verantwortlich. Er entscheidet beim Aufbau dieser Datei, welche Dateien mit group bearbeitet werden sollen. Der Dateityp wird bei (14) QUELL-/ZIELTYP (vor dem Komma)angegeben.

Die Ausgabedateien werden auf der Diskette plaziert, die sich auf dem bei (10) 1. LAUFWERK spezifizierten Laufwerk befindet. Die Namen dieser Ausgabedateien werden aus der Datei uebernommen, deren Name bei (3) GRUPPE spezifiziert ist. Die Ausgabedateien erhalten lediglich einen neuen Dateityp, der bei (14) QUELL-/ZIELTYP (nach dem Komma) definiert wird. Fuer QUELL- und ZIELTYP muessen bei der Anwendung von group unterschiedliche Dateitypbezeichnungen gewaehlt werden. Sind (30) group mit (3) GRUPPE und (26) do mit (1) QUELLE im REDAPROG-Menue aktiviert, hat group Vorrang und do wird ignoriert.

(31) return

Mit dieser Funktion hat man die Moeglichkeit, RETURN-Befehle in REDABAS-Programmen zu belassen oder zu ignorieren. Wird NOreturn gewaehlt, werden alle RETURN-Befehle als Kommentare behandelt. Eine sinnvolle Anwendung dieser Funktion ergibt sich in Verbindung mit der do-Funktion.

(32) comment

Analog wie (31) RETURN kann man mit der comment- Funktion entscheiden, ob Kommentare im Programm belassen oder ausgeblendet werden. Unter Kommentaren werden in diesem Zusammenhang alle mit * beginnenden Befehlszeilen verstanden.

(33) use

Mit USE werden alle REDABAS-Datenbankdateien (Typ.DBD), die in der Eingabedatei eroeffnet werden (d.h. dort mit USE aktiviert werden), in der Reihenfolge ihrer Eroeffnung in der Ausgabedatei (5) STRUK aufgelistet. Bei NOUSE enthaelt die (5) STRUK-Datei nur die Namen der Eingabebefehlsdateien.

4.REDAPROG-Dienste

Aus dem REDAPROG-Menue haben Sie einen Ueberblick ueber die Vielfalt der mit REDAPROG ausfuerbaren Moeglichkeiten und Funktionen erhalten.

Durch richtige Wahl und sinnvolle Verknuepfung der einzelnen Parameter bzw. Einstellungen und Funktionen werden Sie die Datenbankarbeit mit dem Einsatz von REDAPROG verbessern koennen.

Im wesentlichen kristallisieren sich dabei drei Schwerpunkte heraus:

- Laufzeitverkuerzung von REDABAS-Programmen
- Verringerung des Programmieraufwandes
- Vereinfachung der Fehlersuche und Wartung.

Nachfolgend werden Ihnen Mittel und Wege fuer das Erreichen dieser Ziele dargelegt und -wenn erforderlich- an Beispielen demonstriert.

Wir verweisen Sie gleichzeitig auf das Kapitel 7, wo alle REDAPROG-Kommandos in alphabetischer Reihenfolge ausfuerlich beschrieben sind.

4.1. Laufzeitverkuerzung von REDABAS-Programmen

Das Laufzeitverhalten ist ein wichtiges Kriterium fuer die Qualitaet eines Datenbankbetriebssystems. Bereits der richtige Einsatz und die gezielte Auswahl von REDABAS-Befehlen koennen das Laufzeitverhalten positiv beeinflussen.

Wir moechten in diesem Zusammenhang beispielsweise an die Vorteile des Indizierens gegenueber dem Sortieren, an gegebenenfalls satzweise Verarbeitung einer sehr grossen Datei (anstelle des Einsatzes maechtiger Befehle, die sich generell auf die gesammte Datei beziehen), auf das Weglassen von Kommentaren und auf das Verkuerzen von Befehlssteuerwoertern auf vier Buchstaben verweisen.

REDAPROG bietet fuer die Verbesserung des Laufzeitverhaltens von REDABAS-Programmen vor allem die Funktionen do und INCLUDE, deren Einsatz und Wirkung in diesem Zusammenhang nachfolgend beschrieben werden.

Die DO-Funktion

Ausgangspunkt fuer die Anwendung der REDAPROG-do-Funktion ist die Ueberlegung, eine Laufzeitverkuerzung von REDABAS-Programmen durch Eliminieren haeufiger Programeroeffnungen zu erzielen. Damit soll das zeitaufwendige Lesen von der Diskette minimiert werden.

Mehrere zusammengehoerige Befehlsdateien werden in einer groesseren Befehlsdatei zusammengefasst. Diese kann beim Aufruf ganz oder fast vollstaendig in den Arbeitsspeicher geladen werden und ist damit sofort im Zugriff.

Wir moechten Ihnen dies am Beispiel demonstrieren.
Das Programm "menue 2" ruft die folgenden Unterprogramme auf:

```
DO aufnahme
DO rechnung
DO progkun2
```

REDABAS muss bei Ausfuehrung dieses Programms mindestens vier Leseoperationen von der Diskette ausfuehren.

Arbeiten Sie mit REDAPROG, koennen Sie drei dieser Leseoperationen (fuer jedes mit DO <befehlsdatei> eroeffnete Programm) einsparen.

Sie waehlen dazu im REDAPROG-Menue folgende Einstellungen:

```
(1)  QUELLE - MENUE2.PRG
(4)  ZIEL   - MENUE2Z.AUS
(14) QUELL-/ZIELTYP - PRG,AUS
(24) code
(25) display
(26) do
(27) process
(30) NOgroup
```

(In diesen Beispielen moechten wir generell nur die fuer den jeweiligen Anwendungsfall wesentlichen Parameter und Funktionen angeben.)

In die Befehlsdatei "menue2" koennen Sie zusaetzlich vor dem ersten DO-Befehl einfuegen

```
* %ONDO
```

Verlassen Sie das Direktkommando AUSFUEHREN durch druecken von 0<ET>! REDAPROG erzeugt daraufhin die Ausgabedateien entsprechend der aktivierten Funktionen.

Die Funktion (25) display ermoeglicht das Verfolgen der Bearbeitung. Nach Beendigung erscheint wieder das REDAPROG-Menue,Sie geben 99<ET> ein und beenden damit Ihre Arbeit mit REDAPROG.

Mit dem SCP-Kommando TYPE MENUE2Z.AUS koennen sie sich die Ausgabedatei anschauen. Saemtliche Programmaufrufe sind durch

den jeweiligen Programmcode des Unterprogramms ersetzt. Verarbeiten Sie jetzt das REDABAS-Programm MENUE2Z, werden Sie eine deutliche Laufzeitverbesserung feststellen.

Wir moechten Sie auf ein wichtiges Problem beim Anwenden der do-Funktion aufmerksam machen. Die aufgerufenen Unterprogramme enthalten im allgemeinen RETURN-Befehle. Diese Befehle koennen inmitten des zusammengefassten Programms stoeren, da sie zum Programmabbruch fuehren koennen. Deshalb bietet REDAPROG die Moeglichkeit, die RETURN-Befehle der Unterprogramme auszublenden. Sie koennen dafuer die NOreturn-Funktion aktivieren (entweder im Hauptmenue oder durch Einfuegen des Kommandos in die Befehlsdatei).

Sie koennen aber auch jede einzelne Befehlsdatei, die mit RETURN endet, mit dem CODE/NOCODE-Kommando modifizieren:

```
* %NOCODE
RETURN
* %ONCODE
```

Die include-Funktion

Die INCLUDE-Funktion bietet ebenfalls die Moeglichkeit, ein besseres Laufzeitverhalten zu erreichen.

Mit dieser Funktion koennen Programme oder Programmteile, die an sich seperat behandelt werden sollen, in eine "zusammengefuegte" Datei zur effektiveren Abarbeitung gebracht werden. (Es geht dabei prinzipiell um das Reduzieren von Disketten-Leseoperationen wie bei der DO-Funktion.)

An einer Passage des Programms menue2 soll dieser Weg gezeigt werden.

Menue2.PRG

```
      .
      .
CASE auswahl='2'
  DO rechnung
CASE auswahl='3'
      .
      .
```

wird ersetzt durch

MENUE2.PRG

```
      .
      .
CASE auswahl='2'
* %INCLUDE rechnung
CASE auswahl='3'
      .
      .
```

Die Ausgabedatei MENUE2Z.AUS enthaelt die komplette Befehlsfolge von "rechnung".

4.2. Verringerung des Programmieraufwandes

Auch zur Einsparung an Programmieraufwand stellt die Anwendung der INCLUDE-Funktion ein wirksames Mittel dar. Der Grundgedanke dieser Anwendung besteht darin, ausgewaehlte Programme oder Teile von Programmen, die in der gleichen Form haeufig von anderen Programmen genutzt werden (Datenpruefprogramme, Routinen fuer Zeichenkettenmanipulationen, eine Haeufung von STORE-Befehlen, eine Haeufung von Kommentaren usw.) von den "Hauptprogrammen" zu trennen.

Die entsprechende Befehlsfolge wird in eine Datei (Dateityp .INC) gebracht und unabhaengig von den "Hauptprogrammen" behandelt. In diesen "Hauptprogrammen" werden anstelle der Befehlsfolge INCLUDE-Kommandos an den gewuenschten Stellen eingefuegt.

Bevor Sie Ihr REDABAS-Programm dann laufen lassen, wird REDAPROG aktiviert und die INCLUDE-Kommandos werden durch den Programmcode, den die .INC-Datei enthaelt, ersetzt. Zur Kontrolle empfiehlt sich der Ausdruck der Ausgabedatei.

Der Vorteil dieses Verfahrens liegt auf der Hand. Jedes Mal, wenn Sie Ihre .INC-Datei veraendert haben, lassen Sie REDAPROG laufen und alle Befehlsdateien, die diese .INC-Datei verwenden, werden automatisch aktualisiert. Selbstverstaendlich empfiehlt es sich, fuer die Wartung mehrerer Befehlsdateien die do-Funktion oder die group-Funktion (mit GRUPPE-Datei) zu aktivieren.

An einem Beispiel soll die Anwendung der inclide-Funktion in diesem Zusammenhang erlaeutert werden.

- Angenommen, Sie haben vier Befehlsdateien, die eine gleiche Befehlsfolge nutzen:

  DATA1.EIN
  DATA2.EIN
  DATA3.EIN
  DATA4.EIN

- Die von diesen Befehldateien gemeinsam genutzte Befehlsfolge wird in eine Datei gebracht, deren Dateityp .INC ist, z.B. DATPRUEF.INC.

- Anstelle dieser Befehlsfolge fuegen Sie in die Befehlsdateien ein:
  * %INCLUDE DATPRUEF

- Fuer eine spaetere Aktualisierung aller vier Befehlsdateien wird eine GRUPPE-Datei (z. B. NEU.DAT) erstellt.
  Diese Datei wird beispielsweise mit REDABAS erzeugt:
  MODIFY COMMAND NEU.DAT
  (Ausschrift von REDABAS: NEUE DATEI)
  DAT1
  DAT2
  DAT3
  DAT4
  <CTRL W>

- Unterliegt DATPRUEF bestimmten Aenderungen ,veranlassen Sie folgendes:
  Sie starten REDAPROG und tragen im Hauptmenue ein:
  (3) GRUPPE - NEU.DAT
  (14) QUELL-/ZIELTYP - EIN,PRG
  Sie waehlen die Funktion :
  (20) include
  (30) group (Achtung Voreinstellung ist NOgroup!)

- Sie waehlen in der Dialogzeile (0) AUSFUEHREN

- REDAPROG ersetzt die include-Kommandos in allen vier Befehlsdateien durch die geaenderte Befehlsfolge der Datei DATPRUEF.INC

- Es ersetzt folgende aktuallisierte Ausgabedateien:

  DAT1.PRG
  DAT2.PRG
  DAT3.PRG
  DAT4.PRG

  Alle vier Dateien enthalten den vollstaendigen Programmcode.

Wir empfehlen Ihnen fuer die Anwendung der include-Funktion, von jedem Programm zwei Versionen zu behalten:

Einmal das Programm mit den INCLUDE-Kommandos, zum anderen die mit REDAPROG erzeugte Version (analog dem Quell- und Objektprogramm anderer Programmiersprachen). Sie muessen darauf achten, dass beide Versionen nach einer Aktualisierung einander entsprechen.

4.3. Vereinfachung der Fehlersuche und Wartung

Die Moeglichkeit, relativ problemlos Programmierfehler zu erkennen bzw. zu vermeiden und eine leichte Wartung von Programmen setzen eine einwandfreie Programmdokumentation voraus. REDAPROG bietet mehrere Funktionen, die auf die Verbesserung der Dokumentation orientieren.

4.3.1. Die verbesserte Gestaltung der Programmdokumentation

Das eingeben von Programmen am Bildschirm kann sehr aufwendig sein. Um diesen Vorgang so einfach wie moeglich zu gestalten bzw. Formfehler weitgehend auszuschalten, wird der Anwender durch REDAPROG mit den nachfolgend beschriebenen Funktionen unterstuetzt.

Die CONV-Funktion

Die (22) conv-Funktion ermoeglicht, dass Sie beim Eingeben von REDABAS-Steuerwoertern in Ihr Programm nicht auf Gross- oder Kleinschreibung achten brauchen.

Es ist gleichgueltig, ob Sie ihre Programme in Klein- oder Grossbuchstaben (oder gemischt) eingeben, die conv-Funktion bewirkt eine ordnungsgemaesse Ausgabe der Befehlsfolge.

Alle REDABAS-Befehlssteuerwoerter werden in Grossbuchstaben umgewandelt, alle sonstigen Woerter (ausgenommen die in einfachen oder doppelten Anfuerungszeichen stehenden Zeichenreihen) werden klein geschrieben.

Wenn Sie die Befehlssteuerwoerter abgekuerzt haben (REDABAS gestattet das Verkuerzen auf mindestenz vier Buchstaben), werden sie automatisch auf ihre volle Laenge ergaenzt. Die Lesbarkeit von Programmen wird damit deutlich verbessert.

Die skip-Funktion

Mit skip koennen Sie Leerzeilen nach Wunsch in Ihr Programm einfuegen. Sie koennen dies beispielsweise fuer das Kenntlich-machen zusammengehoeriger Befehlsfolgen und zur Gliederung grosser Programme nutzen.

Die page-Funktion

Um den Beginn zusammengehoeriger Befehlsfolgen -z.B. in DO-WHILE- Schleifen eingeschlossene Befehle- zu markieren oder um ein grosses Programm uebersichtlich zu gliedern, koennen Sie mit page einen Seitenvorschub veranlassen.

IF/ENDIF, DO/ENDDO, CASE/ENDCASE-Befehle und der BEFEHLSTABULATOR

Ohne dass Sie eine bestimmte Funktion aktivieren muessen, behandelt REDAPROG Befehlsfolgen, die in IF/ENDIF, DO/ENDDO, CASE/ENDCASE-Klammern eingeschlossen sind, auf folgende Weise:

- Die jeweils zusammengehoerigen Befehlsfolgen werden um die unter (12) BEFEHLTABULATOR definierte Zahl eingerueckt. Anfang und Ende einer zusammengehoerigen Befehlsfolge sind so einfach ueberschaubar
- ENDIF- und ENDDO-Befehle werden um die bei IF oder DO formulierte Bedingung ergaenzt, so dass die Zusammenge-hoerigkeit eindeutig ist.

Beispiel:

Ausschnitt aus einem REDABAS-Programm vor der Bearbeitung mit REDAPROG:

```
     .
     .
     .
* %ONCONV
if #=0
set format to screen
erase
@ 12,5 say 'Kunde nicht vorhanden!'
endif
* %SKIP2
if #>1000
     .
     .
     .
endif
     .
     .
     .
```

nach der Bearbeitung mit REDAPROG:

```
     .
     .
IF #=0
   SET FORMAT TO SCREEN
   ERASE
   @ 12,5 SAY 'Kunde nicht vorhanden!'
ENDIF #=0
            (Es folgen zwei Leerzeilen)

IF #>1000
     .
     .
     .
ENDIF #>1000
```

4.3.2. Das automatische erstellen der Programmdokumentation

Je mehr Programme Sie vorliegen haben, desto schwieriger wird es sein, saemtliche Programme bezueglich Struktur und Inhalt ueberblicken zu koennen. REDAPROG bietet mehrere Varianten fuer die Erstellung einer uebersichtlichen Dokumentation.

Haben Sie verschachtelte Programme, koennen Sie sich mit der STRU-Funktion ein Baumstrukturdiagramm erstellen. Wollen Sie ausserdem wissen, welches Ihrer Programme welche Datenbankdatei verwendet, aktivieren Sie die use-Funktion. Wollen Sie jedoch eine komplette Liste aller Programme haben, realisiert dies die sub-Funktion.

REDAPROG bietet damit differenzierte Moeglichkeiten fuer Ihre Programmdokumentation, die nachfolgend ausfuerlicher erlaeutert werden.

Die Funktionen stru, do und use

Um einen Ueberblick ueber die Baumstruktur verschachtelter REDABAS-Befehlsdateien zu erhalten, nutzen Sie die stru-Funktion in Verbindung mit do.

Dazu muessen Sie die Funktionen (23) stru, (24) code und (26) do aktivieren und einen Namen fuer die Eingabedatei (1) QUELLE und fuer die Ausgabedateien (4) ZIEL und (5) STRUK definieren.

Voraussetzung ist (30) NOgroup und keine Eingabedatei (3) GRUPPE. Als Ergebnis erhalten Sie (ausser der Datei, deren Namen Sie bei (4) ZIEL definiert haben) ein Baumstrukturdiagramm, das den Namen des Hauptprogramms und die Namen aller Unterprogramme enthaelt. Ein Stufenzaehler weist die Programmebene aus.

Wenn Sie zusaetzlich zu den Befehlsdateien die in der jeweiligen Befehlsdatei verwendeten Datenbankdateien mit den zugehoerigen Indexdateien aufgelistet haben moechten, muessen Sie die use-Funktion aktivieren.

Um das Zusammenwirken dieser drei Funktionen zu demonstrieren, moechten wir die REDABAS-Befehlsdatei MENUE2 (vgl. REDABAS-Programmtechnische Beschreibung, Bild 62) als Baumstruktur auflisten.

- Sie waehlen im REDAPROG-Menue

```
( 1) QUELLE - MENUE2.PRG
( 2) WOERTER - RPROGKEY.TXT
( 4) ZIEL - MENUE2Z.AUS
( 5) STRUKTUR - STRUK.STR
(14) QUELL-/ZIELTYP - PRG,AUS
(23) STRU
(24) CODE
(26) DO
(30) NOgroup
(33) USE
```

- Sie aktivieren (0) AUSFUEHREN

- Nach dem REDAPROG-Lauf erhalten Sie ausser der Zieldatei MENUE2Z.AUS die gewuenschte Strukturdatei STRUK.STR mit folgendem Inhalt:

```
0 MENUE2
1       aufnahme
             USE auftrag
1       rechnung
             USE aufpos
             USE bestand INDEX iteilbes
             USE aufpos INDEX inaufpos
             USE rech
             USE auftrag INDEX iaufnrau
        USE aufpos
        USE aufpos INDEX iteilpos
        USE reorg
        USE bestand INDEX iteilbes
        USE bestand INDEX iortbest
1       progkun2
             USE kunden
             USE auftrag INDEX inkunrauf
```

Die SUB-Funktion

Immer dann, wenn Sie eine Liste Ihrer Programme, d. h. die vollstaendigen Befehlsfolgen, benoetigen, koennen Sie mit der (29) sub-Funktion die entsprechenden SCP-Kommandos fuer die Druckausgabe generieren.

REDAPROG erstellt daraufhin eine SCP-SUBMIT-Datei (Dateityp .SUB) und Ihre Programmlisten koennen durch SCP automatisch erzeugt werden.

Im Hauptmenue muessen Sie die Eingabedatei (1) QUELLE, die Ausgabedateien (4) ZIEL, (5) STRUK und die in diesem Fall gewuenschte Ausgabedatei (6) SUBMIT definieren.
Es muss bei (15) eine AUSGABEEINHEIT angegeben werden (Voreinstellung ist LST:). Ausser (29) sub muessn (23) stru, (24) code, (26) do, (29) sub und (30) NOgroup gesetzt sein.

Im Beispiel der Datei MENUE2 saehe dieser Vorgang folgendermassen aus:

- Im REDAPROG-Menue waehlen Sie
  (1) QUELLE - MENUE2.PRG
  (4) ZIEL - MENUE2Z.AUS
  (5) STRUK - STRUK.STR
  (6) SUBMIT - LISTE.SUB
  (15) AUSGABEEINHEIT - PRN:
  (23) STRU
  (24) CODE
  (26) DO
  (29) SUB
  (30) NOGROUP

- Sie aktivieren (0) AUSFUEHREN
  Nach dem REDAPROG-Lauf befindset sich auf der Diskette die Datei LISTE.SUB mit folgendem Inhalt:
  PIP PRN:=MENUE2.PRG
  PIP PRN:=aufnahme.PRG
  PIP PRN:=rechnung.PRG
  PIP PRN:=progkun2.PRG

Um die gewuenschten Programmlisten zu erhalten, geben Sie folgendes SCP-Kommando ein:

SUBM LISTE

Sie erhalten eine Liste saemmtlicher im MENUE2 enthaltenen Befehlsdatein mit Zeilennummern und Vorschub bei jeder Datei. (Moechten sie eine andere Form, waehlen Sie LST anstelle PRN bei (15) AUSGABEEINHEIT).

Wollen Sie nicht alle im MENUE2 enthaltenen Programme auflisten, koennen Sie durch Einfuegen des Kommandos * %onsub bzw. * %nosub in die Eingabedatei eine Auswahl treffen.

4.3.3. Das automatische Erstellen von Sicherheitskopien

Auf die gleiche Weise, wie Sie automatisch Listen erzeugen koennen, ist es moeglich, Sicherheitskopien auf der Diskette zu erzeugen.

Gehen Sie dabei genau so vor wie oben bei der sub-Funktion beschrieben. Sie muessen lediglich im REDAPROG-MENUE bei (15) AUSGABEEINHEIT das Laufwerk spezifizieren, auf dem sich die Diskette befindet, die die Sicherheitskopie aufnehmen soll.

Bleiben wir bei dem Beispiel der Befehlsdatei MENUE2. Waehlen Sie im REDAPROG-MENUE die gleichen Einstellungen wie bei der sub-Funktion im vorigen Abschnitt 3.3.2., aber veraendern Sie:

(6) SUBMIT - BACKUP.SUB
(15) AUSGABEEINHEIT - C

Nach dem REDAPROG-Lauf befindet sich auf der Diskette im - unter (10) genannten- 1. Laufwerk die Datei BACKUP.SUB. Um Ihre Sicherheitskopien zu erhalten, geben Sie jetzt das SCP-Kommando

SUBM BACKUP

Sicherheitskopien koennen auch fuer einzelne Befehlsdateien, nicht zusammengehoerige Gruppen von Befehlsdateien, fuer Baeume oder Teile von Baeumen auf diese unkomplizierte Weise erzeugt werden.

5. Leistungsparameter

Die Arbeit mit REDAPROG setzt die Einhaltung folgender Bedingungen voraus:

Maximale Satzlaenge einer Anweisung (einschliesslich Satzendezeichen hex.0D0A)	100 Zeichen
Maximalzahl der Schluesselwoerter, die In der als Eingabedatei WOERTER spezifizierten Datei vorkommen duerfen.	150
Maximale Laenge dieser Schluesselwoerter	20 Zeichen
Maximale Schachtelungstiefe fuer die Anwendung der DO-Kommandos	8
Maximale Schachtelungstiefe fuer die Anwendung der INCLUDE-Kommandos	8

6. Fehlernachrichten

Fehlernachricht:	Beschreibung:
GLEICHER DATEITYP FUER QUELLE- UND ZIELDATEI	Wenn Sie die group-Funktion nutzen, muessen Sie fuer (14) QUELL-/ZIELTYP unterschiedliche Dateityp-Bezeichnungen waehlen.
FALSCHER DATEITYP	Die bei (14) QUELL-/ZIELTYP angegebenen Dateitypbezeichnungen muessen 3 Zeichen lang sein und untereinander durch ein Komma getrennt sein.
ANWEISUNGSLAENGE IST GROESSER ALS 100	Die Satzlaenge einer Anweisung in der Eingabedatei darf 100 Zeichen nicht ueberschreiten. Die Warnung erscheint auch, wenn aus der Steuerworterweiterung eine Satzlaenge >100 resultiert.
FEHLERHAFTE PROGRAMMSTRUKTUR ******	Ihre Befehlsdatei ****** weist einen logischen Fehler auf. Meist ist ein vergessenes ENDIF oder ENDDO oder ein falsch plaziertes ELSE der Grund.
FALSCHER STRUKTURABSCHLUSS ******	Das vorliegende ENDIF, ENDDO, ENDCASE oder ELSE ist nicht korrekt

CASE SETZT DO-CASE-STRUKTUR VORAUS	Ueberpruefen Sie die DO-CASE-Struktur!
ELSE GEHOERT NICHT ZUM IF	ELSE ist falsch platziert
***** FALSCHES KOMMANDO	Das von ihnen gewaehlte Kommando ***** ist nicht gueltig. Vergleichen Sie die im Kapitel 7 beschriebenen gueltigen REDAPROG-Kommandos!
DAT-DATEI NICHT GEFUNDEN	Die Konfigurationsdatei RPROGKON.DAT befindet sich nicht auf dem aktuellem Laufwerk.
FALSCHE AUSWAHL	Sie haben im REDAPROG-MENUE eine ungueltige Eintragung vorgenommen.
NICHT DEF.DATEI (1), DATEI:******=NUL	Fuer die von Ihnen Gewaehlte Funktion wurde keine Ausgabedatei ****** definiert. (Sie haben beispielsweise die Funktion (23) stru aktiviert und keine Ausgabedatei (5) STRUK definiert.)
WORTAUSWAHL ZU GROSS	Sie duerfen maximal 150 Schluesselwoerter in die bei (2) WOERTER spezifizierte Datei eingeben.
WOERTER NICHT SORTIERT	Die Schluesselwoerter in der bei (2) WOERTER spezifizierten Datei muessen alphabetisch sortiert sein.
***** BZW. y:*****	Die Eingabedatei ***** wurde weder auf dem bei (10) 1.LAUFWERK noch bei (11) 2.LAUFWERK spezifiziertem Laufwerk gefunden.
DATEIENDE (2), DATEI: ******=******	Das Ende der Datei ****** wurde vorzeitig erreicht. Die REDAPROG-Funktion konnte nicht ordnungsgemaess ausgefuert werden, da der auf der Diskette vorhandene Platz fuer Ihre Ausgabedatei nicht ausreicht. (vgl. (10) 1.Laufwerk)

7. REDAPROG-Kommandos

Nachfolgend werden saemtliche REDAPROG-Kommandos - sortiert in alphabetischer Reihenfolge - detailliert beschrieben.

Wir moechten nochmals darauf hinweisen: Sie koennen REDAPROG-Funktionen sowohl im REDAPROG-MENUE als auch in den REDABAS-befehlsdateien aktivieren.

Mit der Uebernahme oder dem Ausserkraftsetzen der im REDAPROG-MENUE angebotenen Funktionen entscheiden Sie zunaechst, wie Sie prinzipiell arbeiten wollen, Sie waehlen eine gewisse Voreinstellung fuer eine automatische Verarbeitung.

Zusaetzlich zu dieser Voreinstellung koennen Sie diese Funktionen mit den nachfolgend beschriebenen Kommandos direkt in Ihren REDABAS-Befehlsdatein aufrufen. Es wird auch Faelle geben, wo sie spezielle Programme oder Programmteile von der voreingestellten Verarbeitung ausklammern wollen. Dann fuegen Sie genau dort in Ihrer REDABAS-Befehlsdatei die gewuenschten Kommandos ein.

Zussammenfassend kann gesagt werden:
Stimmen die Voreinstellwerte im REDAPROG-Menue in jedem Fall mit der gewuenschten Abarbeitung Ihrer REDABAS-Programme ueberein, ist ein zusaetzlicher Aufruf der Kommmandos in den REDABAS-Befehlsdatein wahlfrei.

Wuenschen Sie fuer einzelne mit dem REDAPROG-Lauf zu bearbeitende Programme oder Programmteile eine oder mehrere von den im REDAPROG-Menue voreingestellten Werten abweichende Funktionen, muessen Sie in dieser Form der nachfolgend beschriebenen Kommandos spezifieren.

REDAPROG-Kommandos werden von REDABAS als Kommentare gewertet und beeinflussen damit nicht die REDABAS-Bearbeitung.

REDAPROG-Kommandos haben im allgemeinen folgendes Aussehen:

```
        ON \
* %      > Kommando
        NO /
```

Leerzeichen und % kennzeichnen ein REDAPROG-Kommando.

ON / NO werden verwendet, um eine Funtion zu aktivieren (ON) oder ausser Kraft zu setzen (NO).

Durch das Kommando wird die auszufuehrende Funktion definiert. Einzelne kommandos gestatten zusaetzlich die Angabe eines Parameters.

CODE

Mit diesem Kommando wird aus der im REDAPROG-MENUE unter (1) QUELLE definierte Eingabedatei oder aus den Dateien, deren Namen in der Datei (3) GRUPPE stehen, eine oder mehrere

Ausgabedateien mit vollstaendig aufgelistetem Befehlscode erzeugt.

Ist die do-Funktion aktiv, werden die in der (1) QUELLE-Datei gegebenenfalls vorhandenen Unterprogrammaufrufe ebenfalls mit ihrem Befehlscode aufgelistet. Der Name der erweiterten Ausgabedatei wird bei (4) ZIEL definiert.

Ist die group-Funktion aktiv, werden aus den Dateien, deren Namen (ohne Dateityp !) in der bei (3) GRUPPE spezifizierten Datei stehen, Ausgabedateien mit vollstaendig aufgelistetem Befehlscode erzeugt.

Ein- und Ausgabedateien tragen in diesem Fall den gleichen Namen, unterscheiden sich aber im Dateityp. Dieser wird - unterschiedlich fuer Quelle und Ziel und durch Komma getrennt- bei (14) QUELL- /ZIELTYP angegeben.

Es werden keine Ausgabedateien erzeugt, wenn das Kommando auf NOCODE gesetzt wird.

ACHTUNG! Wenn im REDAPROG-Menue NOCODE gewaehlt wurde, darf die Funktion CODE nicht in den Befehlsdateien durch das Kommando ONCODE aktiviert werden.

Notation	Bedeutung
* %ONCODE	Erzeugen von Befehlscode
* %NOCODE	kein Erzeugen von Befehlscode

Beispiel:

Eingabedatei

```
    .
    .
    .
* %ONCODE
STORE knur TO Skunde
SELECT SECONDARY
FIND &Skunde
* %NOCODE
IF #=0
SET FORMAT TO SCREEN
ERASE
@12,5 SAY 'Kunde nicht vorhanden'
   .
   .
   .
   .
   .
SET TALK ON
* %ONCODE
RETURN
```

Ausgabedatei

```
    .
    .
    .

STORE knure TO skunde
SELECT SECONDARY
FIND &skunde
RETURN
```

COMMENT

Mit dem Kommando COMMENT oder NOCOMMENT kann der Anwender bestimmen, ob im REDABAS-Programm vorhandene Kommentare mit in der mit REDAPROG erstellten Ausgabedatei erscheinen oder nicht. Unter Kommentaren werden in diesem Zusammenhang alle mit * beginnenden Befehlszeilen verstanden.

REDAPROG-Komandos sind davon jedoch ausgenommen. Sie werden in keinem Fall in die Ausgabedatei uebernommen.

Notation	Bedeutung
* %ONCOMMENT	Belassen von Komentaren
* %NOCOMMENT	Entfernen von Kommentaren

Beispiel:

```
Eingabedatei                    Ausgabedatei
    .                               .
    .                               .
    .                               .
* %NOCOMMENT                   CASE auswahl='1'
CASE auswahl='1'                    DO aufnahme
*Neuaufnahme von Kunden         CASE auswahl='2'
DO aufnahme                     *Erstellen von Rechnungen
* %ONCOMMENT                        DO rechnung
CASE auswahl='2'                  .
*Erstellen von Rechnungen         .
DO rechnung                       .
    .
    .
    .
```

CONV

CONV gestattet dem Anwender, seine Programme unabhaengig von GROSS- und KLEINBUCHSTABEN einzugeben.

Bei Anwendung von CONV werden REDABAS-Befehlssteuerwoerter in Grossbuchstaben umgewandelt, alle anderen Woerter werden klein geschrieben.

Eine Ausnahme bilden die in einfachen oder doppelten Anfuehrungszeichen stehenden Zeichenreihen, die original uebernommen werden (z.B. "Geben Sie die Nr. ein:"). Die den Text begrenzenden Anfuehrungszeichen muessen sich auf ein und derselben Zeile befinden, ansonsten wird der Inhalt in Kleinbuchstaben ausgegeben. Dies ist besonders bei mehreren aufeinanderfolgenden Kommentarzeilen zu beachten, die durch die REDABAS-Kommandos TEXT und ENDTEXT eingeschlossen sind.

CONV uebernimmt ausser dem Konvertieren auch das Ergaenzen von verkuerzten Befehlssteuerwoertern auf deren volle Laenge. Die in Grossbuchstaben zu konvertierenden Befehlssteuerwoerter

befinden sich in der unter (2) WOERTER definierten Datei PROGKEY.TXT. Die Befehlssteuerwoerter stehen dort klein geschrieben und in alphabetischer Reihenfolge. Die Datei PROGKEY.TXT kann erweitert oder veraendert werden, wenn diese Bedingungen (klein geschrieben, alphabetisch sortierte Woerter) und die Nichtueberschreitung der Maximalzahl von 150 Woertern eingehalten werden.

Fuer die Wartung von PROGKEY.TXT ist der Nutzer selbst verantwortlich. (Moeglichkeiten dafuer bieten REDABAS, ein Textverarbeitungsprogramm oder der Editor von SCP.)

Sollen andere, vom Anwender gewuenschte Woerter in Grossbuchstaben konvertiert werden, ist dies ebenso problemlos moeglich. Entsprechend muss vom Anwender eine Datei erstellt und gewartet werden unter Einhaltung der fuer PROGKEY.TXT geltenden Bedingungen.

Der Name dieser Datei ist frei waehlbar, der Dateityp muss .TXT sein. Der Dateiname wuerde (anstelle von PROGKEY) unter (2) WOERTER im Redaprog-Menue eingetragen.

Notation	Bedeutung
* %ONCONV	Konvertieren von Woertern
* %NOCONV	Kein Konvertieren

Beispiel:

Eingabedatei:

```
    .
    .
    .
* %ONCONV
erase
set form to print
@ 1,22 say 'Kundenbeleg'
acce 'Bitte Name eingeben' to Sname
* %NOCONV
acce 'Bitte Vorname eingeben' to Svorname
select primary
loca for ...
   .
   .
   .
```

Ausgabedatei:

```
   .
   .
   .
ERASE
SET FORM TO PRINT
@ 1,22 SAY 'Kundenbeleg'
ACCEPT 'Bitte Namen eingeben' TO sname
acce 'Bitte Vornamen eingeben' to Svorname
select primary
LOCATE FOR ...
   .
   .
```

DISPLAY

DISPLAY ermoeglicht die Anzeige der gerade mit REDAPROG bearbeiteten Programmzeilen und der REDAPROG-Nachrichten, nachdem mit (0) AUSFUEHREN die Programmbearbeitung eingeleitet wurde.

Notation	Bedeutung
* %ONDISPLAY	Anzeige
* %NODISPLAY	Unterdruecken der Anzeige

DO
--

Dieses Kommando bewirkt das Ersetzen der in einer REDABAS-Befehlsdatei mit

DO<befehlsdatei>

aktivierten Unterprogrammaufrufe durch den Befehlscode dieser Unterprogramme. Das DO-Kommando bedingt das CODE-Kommando bzw. das Aktivieren der CODE-Funktion im REDAPROG-MENUE.

Die Engabedatei wird bei (1) QUELLE definiert, die erweiterte Ausgabedatei bei (4) ZIEL, der Dateityp beider Dateien ist unter (14) QUELLE-/ZIELTYP anzugeben.

Es gibt zwei Moeglichkeiten, das DO-Kommando auf bestimmte Befehlsdateiaufrufe zu beschraenken. Entweder man setzt das NODO-Kommando oder man nutzt den (16) MAX. DO-STUFE-PARAMETER (vgl. Abschnitt 3.6.4.). Dort kann man die Verschachtelungsebene spezifizieren, bis zu der die Unterprogrammaufrufe durch Befehlscode ersetzt werden sollen.

REDAPROG gestattet maximal 8 Verschachtelungsebenen.

Notation	Bedeutung
* %ONDO	Ersetzen von Unterprogrammaufruf durch Befehlscode
* %NODO	kein "Aufloesen" von Unterprogrammaufrufen

Beispiel:

```
Eingabedatei:                    Ausgabedatei:
    .                                .
    .                                .
    .                                .
* %ONDO
WAIT TO auswahl                  WAIT TO auswahl
DO CASE                          DO CASE
   CASE auswahl = '1'               CASE auswahl = '1'
   DO aufname                         ERASE
* %NODO                              USE auftrag
   CASE auswahl = '2'                APPEND BLANK
   DO rechnung                       @ 2,5 SAY 'Auftr.-Nr.';
   CASE ...                          GET aufnr
    .                                 .
    .                                 .
    .                                 .
                                     CASE auswahl = '2'
                                       DO rechnung
Programm "aufnahme":                 CASE ...
ERASE                                .
USE auftrag                          .
APPEND BLANK                         .
@ 2,5 SAY 'Auftr.-Nr.';
GET aufnr
    .
    .
    .

GROUP
-----
```

Diese Funktion kann nicht ueber ein Kommando innerhalb einer REDABAS-Befehlsdatei aufgerufen werden.

Will man veranlassen, dass die in einer Gruppe zusammengefassten Befehlsdateien mit REDAPROG bearbeitet und einzeln ausgegeben werden, muss man im REDAPROG-Menue die Funktion (30) GROUP aktivieren und eine Datei (3) GRUPPE spezifieren (vgl. Abschnitt 2.3.5.).

INCLUDE

Das INCLUDE-Kommando ermoeglicht das Einfuegen von separat gespeicherten Befehlsfolgen in eine REDABAS-Befehlsdatei. Diese einzufuegende Befehlsfolge muss im Textformat vorliegen und soll den Dateityp .INC haben

Prinzipiell entspricht das INCLUDE-Kommando dem DO-Kommando. Auch bei INCLUDE gibt es einen Parameter, der ueber die maximale Verschachtelungsebene von INCLUDE-Befehlen aussagt. Es sind ebenfalls maximal 8 Verschachtelungsebenen erlaubt.

Notation	Bedeutung
* %INCLUDE <befehlsdatei>	Einfuegen von Befehlsfolgen
* %NOINCLUDE	Ausserkraftsetzen der INCLUDE-Funktion
* %ONINCLUDE	Aktivieren der INCLUDE-Funktion, wenn das Kommando NOINCLUDE gegeben wurde.

Beispiel:

Eingabedatei:

```
    .
    .
    .
STORE 'Kunden' TO datnam
STORE 'DRESDEN' TO ortnam
* %INCLUDE beisp
```

Ausgabedatei:

```
     .
     .
     .
 STORE 'Kunden' TO datnam
 STORE 'DRESDEN' TO ortnam
 USE &datnam
 LIST FOR ort='&ortnam'
```

Datei beisp.INC:

```
USE &datnam
LIST FOR ort='&ortnam
```

Weitere Beispiele siehe Abschnitte 3.1 und 3.2.

PAGE

PAGE veranlasst einen Seitenwechsel. Damit wird dem Nutzer die Moeglichkeit geboten, den Beginn zusammengehoeriger Befehlsfolgen, wie z.B. DO-WHILE-Schleifen, zu markieren oder sehr grosse Programme zu gliedern.

Notation	Bedeutung
* %PAGE	Erzeugen von Seitenvorschub
* %NOPAGE	Unterdruecken von Seitenvorschub
* %ONPAGE	Aktivieren der PAGE-Funktion, wenn das Kommando NOPAGE gegeben wurde.

PROCESS

PROCESS bewirkt die Abarbeitung der REDAPROG-Kommandos. Will man bestimmte Programme oder Programmteile von einer REDAPROG-Bearbeitung ausklammern, setzt man das Kommando NOPROCESS. Die dem Kommando NOPROCESS folgenden Befehlszeilen in der Eingabedatei werden unveraendert in die Ausgabedatei uebernommen.

Notaion	Bedeutung
* %ONPROCESS	Abarbeitung von REDAPROG-Kommandos
* %NOPROCESS	Unterdruecken der Abarbeitung von REDAPROG-Kommandos

Beispiel:

Eingabedatei:

```
    .
    .
    .
* %ONCONV
* %ONPROCESS
case auswahl = '3'
erase
use aufpos
index on teilnr to iteilpos
* %NOPROCESS
use aufpos index iteilpos
total on teilnr to reorg
    .
    .
    .
```

Ausgabedatei:

```
        .
        .
        .
CASE auswahl = '3'
 ERASE
 USE aufpos
 INDEX ON teilnr TO iteilpos
use aufpos index iteilpos
total on teilnr to reorg
          .
          .
          .
```

RETURN

REDABAS-Return-Befehle in Befehlsdatein koennen im REDAPROG-Lauf bei Anwendung der DO-Funktion stoeren. Ruft beispielsweise eine Befehlsdatei weitere Befehlsdateien (mit DO <befehlsdatei>) auf und wird die do-Funktion aktiviert, koennen die RETURN-Befehle, die die einelnen Befehlsdateien abschliessen, zum Programmabbruch fuehren.

REDAPROG bietet aus diesem Grund die Moeglichkeit zu entscheiden, ob RETURN-Befehle beibehalten [(31)RETURN oder Kommando ONRETURN] oder ausgeblendet (NORETURN) werden sollen.

Entscheidet man sich fuer NORETURN, ignoriert REDAPROG in REDABAS-Befehlsdateien enthaltene RETURN-Befehle und kennzeichnet sie als Kommentare, unabhaengig davon, ob das COMMENT-Kommando gegeben wurde oder nicht.

Notation	Bedeutung
* %ONRETURN	Verarbeiten von REDABAS-RETURN-Befehlen
* %NORETURN	Ignorieren von REDABAS-RETURN-Befehlen

Beispiel:

```
Eingabedatei                    Befehlsdatei "aufnahme"

    .                           ERASE
    .                           USE auftrag
    .                           APPEND BLANK
* %ONDO                         @ 2,5 SAY 'Auftr.-Nr,' GET;
* %NORETURN                     aufnr
DO CASE                         @ 3,5 SAY 'Kunden-Nr.' GET;
CASE auswahl = '1'              kunr PICTURE 'xxx'
   DO aufnahme                  @ 4,5 SAY 'DATUM' GET datum;
CASE auswahl = '2'              PICTURE '99.99.99'
   DO rechnung                  READ
       .                        RETURN
       .
       .

 Ausgabedatei
      .
      .
      .
DO CASE
   CASE auswahl = '1'
   * * -- DO aufnahme -- * * * *

   ERASE
   USE auftrag
   APPEND BLANK
   @ 2,5 SAY 'Auftr.-Nr.' GET aufnr
   @ 3,5 SAY 'Kunden-Nr.' GET kunr PICTURE 'xxx'
   @ 4,5 SAY 'Datum' GET datum PICTURE '99.99.99'
   READ
   * *  RETURN
   CASE auswahl = '2'
   ERASE
   SELECT SECONDARY
       .
       .
       .
SKIP
----
```

SKIP <anzahl> veranlasst Zeilenvorschub in der bei <anzahl> angegebenen Groesse in der Ausgabedatei. Damit bietet das SKIP-Kommando -analog zum PAGE-Kommando- die Moeglichkeit zur Gliederung von Programmen.

Notation	Bedeutung
* %SKIP<anzahl>	Erzeugen von Zeilenvorschub
* %NOSKIP	Unterdruecken von Zeilenvorschub
* %ONSKIP	Aktivieren der SKIP-Funktion, wenn das Kommando NOSKIP gegeben wurde

Beispiel:

* %SKIP2	Es werden 2 Leerzeilen ausgegeben.

SUB

Das SUB-Kommando veranlasst das Erzeugen einer SCP-SUBMIT-Datei (Dateityp.SUB) fuer die Anwendung des SCP-Kommandos SUBM. Es koennen je nach Wunsch Befehlsdateien erstellt werden (vgl. Abschnitte 4.3.2. und 4.3.3.).

Die Anordnung der einzelnen Befehlsdateien innerhalb der SUBMIT-Datei erfolgt in der Reihenfolge ihres Aufrufs in der Eingabedatei. Die Eingabedatei wird bei (1) QUELLE spezifiziert. Die gewuenschte Ausgabedatei, d. h. die SCP-SUBMIT-Datei wird bei (6) SUBMIT spezifiziert.

Voreinstellung ist LISTE.SUB. Ausserdem muessen die Dateien (4) ZIEL und (5) STRUK definiert sein.

SUB setzt das Aktivieren der Funktionen (23)stru, (24)code, (26)do und (30)NOgroup bzw. die entsprechenden Kommandos voraus. Es muss bei (15) AUSGABEEINHEIT der logische SCP-Geraetename spezifiziert werden.

Voreinstellung mit LST:, d. h. Ausgabe auf dem Drucker. (Weitere Moeglichkeiten sind andere logische Geraetenamen, die das Dienstprogramm PIP des SCP zulaesst wie z. b. PRN oder auch die Bezeichnung eines Laufwerkes, wenn Sie mit dem SUB-Kommando Sicherheitskopien erstellen wollen.)

ACHTUNG ! Wenn im REDAPROG-Menue NOSUB gewaehlt wurde, darf die Funktion SUB nicht in den Befehlsdateien durch das Kommando ONSUB aktiviert werden.

Notation	Bedeutung
* %ONSUB	Erzeugen der SUBMIT-Datei
* %NOSUB	kein erzeugen der SUBMIT-Datei

Beispiel:
Eingabedatei: MENUE2.PRG

Ausgabedatei:
```
LISTE.SUB
PIP LST:=MENUE2.PRG
PIP LST:=aufnahme.PRG
PIP LST:=rechnung.PRG
PIP LST:=progkun2.PRG
```

STRU

STRU veranlasst das Erzeugen einer Struktur-Datei, die die Baumstruktur der REDABAS-Befehlsdateien enthaellt. Die jeweilige Programmebene zeigt ein Stufenzaehler an.

Die Eingabedatei wird bei (1) QUELLE spezifiziert. Der Name der Strukturdatei (Ausgabedatei) wird bei (5) STRUK definiert, Voreinstellung ist SRUK.STR. Ausserdem muss die Datei (4)

ZIEL definiert sein.

Vorraussetzung fuer das Erstellen der Strukturdatei ist, dass -ausser STRU- die DO- und CODE-Funktion aktiviert sind, NOgroup gesetzt wurde und keine (3) GRUPPE-Datei existiert.

Wenn ausserdem die USE-Funktion gewaehlt wurde, werden zusaetzlich zu den Befehlsdateien auch die Datenbankdateien in der Reihenfolge ihres Aufrufs in der Strukturdatei aufgelistet (vgl. Abschnitt 4.3.2.).

ACHTUNG! Wenn im REDAPROG-Menue NOSTRU gewaehlt wurde, darf die Funktion STRU nicht in den Befehlsdateien durch das Kommando ONSTRU aktiviert werden.

Notation	Bedeutung
* %ONSTRU	Erzeugen der Strukturdatei
* %NOSTRU	kein Erzeugen einer Strukturdatei

Beispiel:

Eingabedatei:	Ausgabedatei
MENUE2.PRG	STRUK.STR
(es wurde NOUSE gewaehlt)	0 MENUE2
	1 aufnahme
	1 rechnung
	1 progkun2

USE

Das USE-Kommando bewirkt das Auflisten der Namen der REDABAS-Datenbankdateien, die von der Eingabedatei genutzt werden, in der bei (5) STRUK definierten Strukturdatei.

Die Namen der Datenbankdateien werden in der Reihenfolge angefuehrt, in der sie von den jeweiligen Befehlsdateien aktiviert werden.

Sind Indexdateien vorhanden, werden auch diese mit aufgelistet.

Notation	Bedeutung
* %USE	Auflisten der Namen von Datenbank-dateien und Befehlsdateien in der Struktur-Datei
* %NOUSE	Die Strukturdatei enthaelt nur die Namen der Befehlsdateien.

Beispiel:

```
0 MENUE2
1       aufnahme
             USE auftrag
1       rechnung
             USE aufpos
             USE bestand INDEX iteilbes
             USE aufpos INDEX inaufpos
             USE rech
             USE auftrag INDEX iaufnrau
        USE aufpos
        USE aufpos INDEX iteilpos
        USE reorg
        USE bestand INDEX iteilbes
        USE bestand INDEX iortbest
1       progkun2
             USE kunden
             USE auftrag INDEX inkunrauf
```

Sachwortverzeichnis

Anwenung 5 f.
Ausgabedateien 11, 14, 16 f, 19
AUSGABEEINHEIT 16, 28, 29

Beenden (von REDAPROG) 10
BEFEHLSTABULATOR 16, 25
Betriebssystem 6, 10

code 18
CODE 32 f.
comment 19
COMMENT 34
conv 17, 25
CONV 34
Cursoradressierung 8, 9

Dateinamen 12
Dateityp 13
Datenbankdatei 20
Dienste (von REDAPROG) 20
Display 18, 21
DISPLAY 36
do 18, 21, f., 27
DO 16, 30, 36

Eigeninstallation 8 f.
Eingabedatei 13, 19
Einleitungszeichen 8 f

Fehlernachrichten 9, 30
Funktionen 12, 17, ff., 25

group 15, 19, 33
GROUP 37
GRUPPE 13

Handhabung 6 ff.

include 17, 22, 23
INCLUDE 17, 23, 24, 30, 37
Installieren 6 ff.

LAUFWERK 15
Laufzeitverkuerzung 20, 21, 22
Leistungsparameter 30

MAX. DO-STUFE 30
MAX. INCLUDE-STUFE 30
Menue 11

page 18, 25
PAGE 38
process 18
PROCESS 38

Programmdokumentation 24, 27

QUELLE 13, 27, 33
QUELL-/ZIELTYP 13, 15, 19, 27,3 6

REDABAS 11, 20, 21, 23, 25, 32, 35
return 19
RETURN 22, 39
RPROGINS 7
RPROGINS.COM 5, 6
RPROGKEY.TXT 5,
RPROGKON.DAT 5, 7

Schluesselwoerter 30
Sicherheitskopie 6, 29
Sichern 12
skip 17, 25
SKIP 40
Steuercodes 7 f.
sub 18, 28
SUB 41
SUBM 29
SUBMIT 14, 28, 29, 41
Starten 10
stru 18, 27
STRU 41
STRUK 14, 27
STRUKTURTAB. 16

Umgebung 15
use 20, 27
USE 42

WOERTER 13, 30, 34, f.

ZIEL 14, 27, 33